

GFD-R-P.xxx
Category: Recommendation
GLUE Working Group
<http://redmine.ogf.org/projects/glue-wg>

Authors:
Warren Smith[°], Texas Advanced
Computing Center
David Meredith[°], STFC
Shiraz Memon*, FZJ
JP Navarro*, University of Chicago
*co-chairs
[°]editors

November, 2014

GLUE v. 2.0 – Reference Realization to JSON Schema

Status of This Document

This document is under discussion in the Open Grid Forum GLUE Working Group.
Distribution is unlimited.

Copyright Notice

Copyright © Open Grid Forum (2014). All Rights Reserved.

Abstract

The GLUE 2 specification [1] is an information model for Grid entities described in natural language enriched with a graphical representation using UML Class Diagrams. This document presents a realisation of this information model as JSON [2] and formally describes this realization using JSON Schema [4].

Contents

1.	About this Document	4
2.	Notational Conventions	4
3.	JSON	4
4.	JSON Schema	4
5.	JSON Schema Realisation	4
5.1	Namespaces	4
5.2	Schema as a Set of JSON Documents	5
5.3	GLUE 2 JSON Flat Document Structure	6
5.4	Representing GLUE 2 Entities in JSON	8
5.4.1	Associations	8
5.4.2	Enumerations	9
5.5	Extending the GLUE 2 JSON Schema	10
6.	Security Considerations	10
7.	Author Information	11
8.	Contributors & Acknowledgements	11
9.	Intellectual Property Statement	11
10.	Disclaimer	11
11.	Full Copyright Notice	13
12.	References	14
13.	Appendix	15
13.1	AccessPolicy.json	15
13.2	Activity.json	15
13.3	AdminDomain.json	16
13.4	ApplicationEnvironment.json	17
13.5	ApplicationHandle.json	18
13.6	Benchmark.json	19
13.7	ComputingActivity.json	19
13.8	ComputingEndpoint.json	22
13.9	ComputingManager.json	23
13.10	ComputingService.json	25
13.11	ComputingShare.json	26
13.12	Contact.json	29
13.13	DataStore.json	30
13.14	Domain.json	30
13.15	Endpoint.json	31
13.16	Entity.json	33
13.17	ExecutionEnvironment.json	34
13.18	Glue2.json	36
13.19	Location.json	40
13.20	Manager.json	41
13.21	MappingPolicy.json	42
13.22	Policy.json	42
13.23	Resource.json	43
13.24	Service.json	43

13.25	Share.json	44
13.26	StorageAccessProtocol.json.....	45
13.27	StorageEndpoint.json.....	46
13.28	StorageManager.json	46
13.29	StorageService.json	46
13.30	StorageServiceCapacity.json	47
13.31	StorageShare.json	47
13.32	StorageShareCapacity.json.....	49
13.33	ToComputingService.json.....	49
13.34	ToStorageService.json	50
13.35	UserDomain.json	50

1. About this Document

The GLUE 2.0 Information model defined in [1] is a conceptual model of Grid entities. In order to be adopted by technology providers, a realisation in terms of a concrete data model is needed. This document provides the normative realisation of the GLUE 2.0 conceptual model in terms of a JSON Schema [4,5]. The document also elaborates on the design choices adopted to map the entities and relationships of the conceptual model into the concrete data model.

This document is maintained by OGF's GLUE Working Group, which signs responsible for documenting errata and releasing revisions as defined by the OGF document process. Errors and feedback in general should be directed to the GLUE WG mailing list, at <glue-wg@ogf.org>.

2. Notational Conventions

The key words 'MUST,' "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" are to be interpreted as described in [rfc -2119].

3. JSON

JSON is a lightweight textual format for exchanging structured information that was originally derived from JavaScript. However, JSON is independent of JavaScript and its use has spread widely. JSON is formally defined in [3].

JSON is a simple language with only two complex types: Objects that specify key/value pairs and arrays that specify ordered lists of values. The values in these complex types can themselves be complex types or they can be a string or a number. This simple structure makes it very easy to map JSON documents to structures in many different programming languages.

4. JSON Schema

JSON was defined and became popular before it had a formal language to describe the structure of specific classes of JSON documents. An effort called JSON Schema was formed to address this by submitting several draft versions of their schema specification to the IETF. The current version at the time of this writing is draft 04 [5,6] and this document uses that version.

A JSON Schema document constrains the hierarchy of a conforming JSON (instance) document in terms of objects and arrays. It can also describe the required names in objects, the types of values in object attribute values, the types of values in arrays, and constrain possible values (e.g. a value must be from a set of possible values). In addition, JSON Schema allows human-readable descriptions to be embedded into the schema specifications, allowing a schema author to provide in-line documentation.

There are a variety of JSON Schema validators available in different programming languages (see the JSON Schema website for suggestions). This allows projects to add JSON Schema validation to their software infrastructure if they so desire.

5. JSON Schema Realisation

5.1 Namespaces

The Open Grid Forum has provided guidelines for uniquely identifying uniform names in the GGF/OGF domain [ogf-ns]. Based on this document, we have adopted the following

namespace for the JSON Schema realization of GLUE 2.0:

GLUE-JSON-NS ::= 'http://schemas.ogf.org/glue/' YYYY '/' MM '/spec_' M.N '_r' R '/' ENTITYNAME'.json'

YYYY: year of the normative document of the GLUE specification

MM: month of the normative document of the GLUE specification

M.N: M is the major version and N is the minor version of the GLUE conceptual model

R: component to be used to specify the revision number of the JSON realization; this number SHOULD be incremented each time a new non-backwards compatible version is published

ENTITYNAME: represents the name of the GLUE entity in the JSON document; every conceptual entity/class has corresponding JSON realisation

As a non-normative example, the namespace for the first release of the JSON document for Entity of the final GLUE 2.0 specification [glue-2] is:

http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json

5.2 Schema as a Set of JSON Documents

The GLUE2 JSON Schema is specified using a set of schema documents [7] with each GLUE 2 Entity having its own document. JSON Schema allows schema documents to be composed. We took this approach for easier comprehension and because separate entity schema documents may be of use to those using document-oriented databases to store GLUE 2 JSON documents.

The **Glue2.json** schema document is the aggregating root document containing the major entities, and should be used to validate full GLUE 2 JSON documents. This document, partially shown in Figure 1, defines an object with keys that are named after the GLUE 2 Entities and values that are arrays of GLUE 2 Entity objects. These Entity objects are referred to by id and are specified in separate JSON Schema files. All of the entities are arrays because even though some Entities occur at most once per site or per grid, a GLUE 2 document could describe multiple sites or grids.

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0/Glue2.json",
  "description": "A GLUE 2 document",
  "type": "object",
  "properties": {
    "Entity": {
      "type": "array",
      "items": {
        "type": "object",
        "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0/Entity.json"}]
      }
    },
    "Location": {
      "type": "array",
      "items": {
        "type": "object",
        "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0/Location.json"}]
      }
    },
    ...
  }
}
```

Figure 1. Partial Glue2.json top-level schema definition file.

5.3 GLUE 2 JSON Flat Document Structure

The top-level schema approach described above leads to a flat document structure where the different GLUE 2 Entities are equal siblings defined at the same level of nesting. This is similar to the structure adopted in the GLUE 2 XML rendering [9]. There are a number of functional justifications for adopting a flat document structure compared to deeply nesting entities into parent-child relationships. These are fully explained in the GLUE 2 XML rendering document [9] and will only be summarised here:

- The GLUE 2 information model is not a strict tree structure. Entities with multiple parents and many-to-many relationships cannot be represented by nesting alone. Conversely, representing associations using ID references alone produces no such limitations.
- Nesting entities defines a strong parent-child relationship, where the lifespan of the child is strongly linked to that of the parent (UML composition). However, the entity relationships in GLUE 2 represent weaker UML aggregations where entities can exist in isolation with their own independent life spans. This is better suited to the ID referencing.
- Data can be 'sliced' more easily for use in projection queries (i.e. queries that require that only selected entities are returned).
- The loose coupling provided by ID references allows query results to be split across multiple JSON documents. For example, one document could provide a list of service endpoints whilst another could provide contact information. This allows potentially sensitive information to be split into different documents.

A partial example of this flat structure is shown below with full examples available at [8].

```
{
  "ComputingEndpoint": [
    {
      "Associations": {
        "ShareID":
["urn:glue2:ComputingShare:development.stampede.tacc.xsede.org",
        ...],
        "ServiceID": "urn:glue2:ComputingService:stampede.tacc.xsede.org"
      },
      "Capability": [
        "executionmanagement.jobdescription",
        "executionmanagement.jobexecution",
        "executionmanagement.jobmanager"
      ],
      "CreationTime": "2013-05-05T22:24:03Z",
      "HealthState": "unknown",
      "ID": "urn:glue2:ComputingEndpoint:gram5-
slurm.stampede.tacc.xsede.org",
      "ImplementationName": "gram5",
      "ImplementationVersion": "5.2.3",
      "Implementor": "The Globus Alliance",
      "InterfaceName": "globus.gram5",
      "Name": "gram5-slurm",
      "QualityLevel": "production",
      "ServingState": "production",
      "Technology": "legacy",
      "URL": "login5.stampede.tacc.utexas.edu:2119/jobmanager-slurm",
      "Validity": 3600
    },
    ...
  ],
  "ComputingManager": [
    {
      "Associations": {
        "ResourceID":
```

```
[{"urn:glue2:ExecutionEnvironment:NodeType1.stampede.tacc.xsede.org"},
  {"ServiceID": "urn:glue2:ComputingService:stampede.tacc.xsede.org"}
]
  "CreationTime": "2013-05-05T22:24:05Z",
  "Homogeneous": false,
  "ID": "urn:glue2:ComputingManager:stampede.tacc.xsede.org",
  "Name": "SLURM",
  "ProductName": "SLURM",
  "Reservation": true,
  "SlotsUsedByLocalJobs": 92784,
  "TotalLogicalCPUs": 103040,
  "TotalPhysicalCPUs": 12880,
  "TotalSlots": 103040
}
],
"ComputingService": [
  {
    "Associations": {
      "ContactID": [],
      "LocationID": "urn:glue2:Location:TexasAdvancedComputingCenter",
      "ServiceID": []
    },
    "Capability": [
      "executionmanagement.jobexecution",
      "executionmanagement.jobdescription",
      "executionmanagement.jobmanager",
      "executionmanagement.executionandplanning",
      "executionmanagement.reservation"
    ],
    "CreationTime": "2013-05-05T22:24:06Z",
    "ID": "urn:glue2:ComputingService:stampede.tacc.xsede.org",
    "Name": "SLURM",
    "PreLRMSWaitingJobs": 0,
    "QualityLevel": "production",
    "RunningJobs": 403,
    "StagingJobs": 0,
    "SuspendedJobs": 0,
    "TotalJobs": 552,
    "Type": "ipf.SLURM",
    "WaitingJobs": 149
  }
],
"ComputingShare": [
  {
    "Associations": {
      "ResourceID": [],
      "ServiceID": "urn:glue2:ComputingService:stampede.tacc.xsede.org"
    },
    "CreationTime": "2013-05-05T22:24:04Z",
    "ID": "urn:glue2:ComputingShare:development.stampede.tacc.xsede.org",
    "LocalRunningJobs": 14,
    "LocalSuspendedJobs": 0,
    "LocalWaitingJobs": 0,
    "MappingQueue": "development",
    "MaxSlotsPerJob": 152,
    "MaxWallTime": 28800,
    "Name": "development",
    "RequestedSlots": 0,
    "RunningJobs": 14,

    "ServingState": "production",
    "SuspendedJobs": 0,
    "TotalJobs": 14,
    "UsedSlots": 400,
    "WaitingJobs": 0
  },
  ...
],
```

```

"Location": [
  {
    "Country": "US",
    "CreationTime": "2013-05-05T22:24:03Z",
    "ID": "urn:glue2:Location:TexasAdvancedComputingCenter",
    "Latitude": 30.390132,
    "Longitude": -97.725309,
    "Name": "Texas Advanced Computing Center",
    "Place": "Austin",
    "Validity": 86400
  }
]
}

```

Figure 2. Example GLUE2 document (condensed).

5.4 Representing GLUE 2 Entities in JSON

As shown in the example above, each GLUE 2 entity is mapped to a JSON object using the same level of nesting within the document. The keys of an entity object match the names of attributes defined for that entity in the GLUE 2 specification.

The values in the entity objects are based on the GLUE 2 specification. If an entity attribute can have at most one value, a simple value is used in the JSON document. If an entity attribute can have more than one value, an array is used in the JSON document, even if only one value is present in a specific document.

Two important details to clarify are how associations and enumerations are represented.

5.4.1 Associations

In the conceptual model, several association types are represented. They can be classified in terms of the multiplicity (one-to-one, one-to-many), in terms of the navigability (bi-directional, unidirectional) or in terms of the association type (binary, aggregation, composition, association class).

Associations between entities are represented using ID keys and references to those IDs. Each entity has a unique ID specified in the 'ID' key. Associations are then specified by referencing the IDs of other entities inside an 'Associations' object. For example, in Figure 2 above, the ComputingService has an ID of 'urn:glue2:ComputingService:stampede.tacc.xsede.org' and the ComputingShare references this value in the ServiceID key. This means that the ComputingShare is associated with the specific ComputingService. Note that the keys used to refer to other entities have names that end in the suffix 'ID', such as 'ServiceID' in this example.

Following the pattern established in the GLUE 2 XML rendering [9], entities in the JSON schema reference other entities using keys named after the parents of the referenced entities, rather than defining more specialized keys named directly after the referenced entity. For example, we use 'EndpointID' to reference a computing endpoint and not 'ComputingEndpointID' (in this case, Endpoint is the parent of ComputingEndpoint so we use 'EndpointID'). This caters for a well-defined use case: standard queries can be executed on a document irrespective of the subtype of the referenced entity (i.e. facilitating a polymorphic query).

The Schema allows associations to be represented bidirectionally. For example, as shown above, a ComputingEndpoint can reference its computing service using a 'ServiceID' key. On the other side of the relationship, the ComputingService references its computing endpoints using an EndpointID key. This bidirectional representation can be convenient in some situations and inconvenient in other situations.

To ensure that every association is represented, at least one direction of each association is required in the schema. This approach also makes interoperability easier because a consumer can look for the association direction that the schema requires.

The directions that are required were selected using the heuristics:

- In the case of a many-to-one association, “the many” refer to “the one”.
- In the case of a one-to-one association:
 - The entity that is created/deleted more often refers to the entity that is created/deleted less often.
 - If the entities that are created/deleted at the same rate, they refer to each other in a bidirectional manner.

Examples of these recommendations in practice are shown in Figure 2.

Many ComputingShares refer to one ComputingService, so the ComputingShares are required to use their ServiceID to refer to the ID of the ComputingService.

A Location is rarely deleted (or never is), so the ComputingService refers to the Location using a LocationID attribute.

The ComputingManager and ComputingService entities are created and deleted at the same rate, so the ComputingManager refers to the ComputingService using a ServiceID attribute and the ComputingService refers to the ComputingManager using a ManagerID attribute.

It can occur that an association direction is required, but there is no association to represent. In this case, the value of the association is either the JSON null value (when the association is one-to-one) or an empty array (when the association is one to many). The empty arrays under Associations in Figure 2 are an example of this.

5.4.2 Enumerations

In the conceptual model, the enumerations are defined using a set of strings. They are categorised as: ‘open’ enables applications to append the existing enumerations, ‘closed’ restricts the use of the existing enumerations.

Closed enumerations are specified using the JSON schema `enum` keyword. Figure 3 shows the `QualityLevel` enumeration from the `Service.json`. JSON schema validators will ensure that the value provided for this key in the JSON document is one of the strings specified in the enumeration.

```

"Service": {
  ...
  "QualityLevel": {
    "type": "string",
    "description": "QualityLevel_t",
    "enum": ["development", "pre-production", "production", "testing"]
  },
  ...
}

```

Figure 3: Example of a close enumeration

One of our design choices is to embed the closed enumerations into the entities instead of defining them separately and referring to them by id. We choose this approach to improve readability of the schemas. A few of the enumerations are used by more than one entity which leads to multiple places to update when an enumeration changes, but the values in the closed enumerations rarely change. Alternative options are:

- Define closed enumerations separately and refer to them in entities
 - This hurts readability
- Embed closed enumerations that are only used once, refer to separately defined enumerations if they are used by several entities
 - This is an inconsistent method and would also hurt readability

For the open enumerations, there is no JSON schema validation that can be performed because any value is valid. In this case, the value is a simple string and the description

associated with it refers to the name of the open enumeration type. The recommended values for the enumerations are given in the GLUE v 2.0 specification. Figure 4 shows how the Type attribute of a Service is specified as a string, but the human-readable description references the open enumeration ServiceType_t. Section B.31 of the GLUE v 2.0 specification lists the set of values that have already been defined for this enumeration.

```
"Service": {  
  ...  
  "Type": {  
    "type": "string",  
    "description": "ServiceType_t (open enumeration)"  
  },  
  ...  
}
```

Figure 4: An open enumeration

5.5 Extending the GLUE 2 JSON Schema

By default, additional properties are allowed in JSON schema. Extension could be prevented by specifying 'additionalProperties: false' so that only the keys specified for an object would be allowed. If any other keys were present, validation would fail. However, the value of additionalProperties cannot be overridden by an extending JSON schema object and so we do not use this feature as we wish to support extension.

During document validation, it is important to note the following; JSON schema validators only check that required keys are present and that their values are valid against the schema. If a JSON object has keys that aren't specified in its corresponding object schema, they are simply ignored.

A provider of a GLUE 2 JSON document MAY include additional, arbitrary JSON content in the Extension object that are present in all entities. This content will not be validated against a schema.

A provider of a GLUE 2 JSON document SHOULD NOT include additional, arbitrary JSON content outside of Extension. As described above, this JSON schema ignores additional object attributes.

If a provider wishes to extend the GLUE 2 JSON Schema, they should define their own entities that extend the entities described in the schema and then define their own variant of Glue2.json that includes their entities along with unmodified GLUE 2 JSON Schema entities.

6. Security Considerations

Security considerations related to the GLUE information model are discussed in section 9 of the GLUE specification [glue-2]. Additional considerations apply to the use of XML – for those, see [rfc-3470].

7. Author Information

Warren Smith
Texas Advanced Computing Center
Research Office Complex 1.101
J.J. Pickle Research Campus, Building 196
10100 Burnet Road (R8700)
Austin, Texas 78758-4497
E-mail: wsmith@tacc.utexas.edu

David Meredith
Scientific Computing Department
Science and Technology Facilities Council
Daresbury Laboratory
Warrington, Cheshire, WA4 4AD
E-mail: david.meredith@stfc.ac.uk

Shiraz Memon
Jülich Supercomputing Centre (JSC)
Wilhelm-Johnen-Straße
52425 Jülich, Germany
Email: a.memon@fz-juelich.de

John-Paul Navarro
University of Chicago/Argonne National Laboratory
Mathematics & Computer Science Division, Building 221
9700 S. Cass Avenue
Argonne, IL 60439 (USA)
E-mail: navarro@mcs.anl.gov

8. Contributors & Acknowledgements

We gratefully acknowledge the contributions made by the members of the GLUE working group.

9. Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights, which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

10. Disclaimer

This document and the information contained herein is provided on an "As Is" basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that

the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

11. Full Copyright Notice

Copyright (C) Open Grid Forum (2014). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included as references to the derived portions on all such copies and derivative works. The published OGF document from which such works are derived, however, may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing new or updated OGF documents in conformance with the procedures defined in the OGF Document Process, or as required to translate it into languages other than English. OGF, with the approval of its board, may remove this restriction for inclusion of OGF document content for the purpose of producing standards in cooperation with other international standards bodies.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

12. References

- [1] S. Andreozzi, et. al. GLUE specification v 2.0. GFD.147.
<http://www.ogf.org/documents/GFD.147.pdf>
- [2] JSON (JavaScript Object Notation). <http://www.json.org/>
- [3] D. Crockford. The application/json Media Type for JavaScript Object Notation (JSON). IETF RFC 4627.
- [4] JSON Schema. <http://json-schema.org/>
- [5] JSON Schema: core definitions and terminology draft-zyp-json-schema-04. <http://json-schema.org/latest/json-schema-core.html>
- [6] JSON Schema: interactive and non-interactive validation draft-zyp-json-schema-04. <http://tools.ietf.org/html/draft-zyp-json-schema-04>
- [7] GLUE 2 JSON Schema. <https://github.com/OGF-GLUE/JSON/tree/master/schema>
- [8] GLUE 2 JSON examples. <https://github.com/OGF-GLUE/JSON/tree/master/examples>
- [9] GLUE 2 XML Rendering. <http://www.ogf.org/documents/GFD.209.pdf>

13. Appendix

Schema available at [\[TODO – add link of OGF schema\]](#) and examples available in github: <https://github.com/OGF-GLUE/JSON>

13.1 AccessPolicy.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/AccessPolicy.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Policy.json"}],
  "properties": {
    "Associations": {
      "type": "object",
      "properties": {
        "EndpointID": {
          "type": "string",
          "description": "The ID of the Endpoint this AccessPolicy is for"
        }
      },
      "required": ["EndpointID"]
    }
  },
  "required": ["Associations"]
}
```

13.2 Activity.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Activity.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Associations": {
      "type": "object",
      "properties": {
        "UserDomainID": {
          "type": ["string", "null"],
          "description": "The ID of the UserDomain managing this Activity"
        },
        "EndpointID": {
          "type": ["string", "null"],
          "description": "The ID of the Endpoint managing this Activity"
        },
        "ShareID": {
          "type": ["string", "null"],
          "description": "The ID of the Share servicing this Activity"
        },
        "ResourceID": {
          "type": ["string", "null"],
          "description": "The ID of the Resource executing this Activity"
        },
        "ActivityID": {
          "type": "array",
          "description": "The IDs of other Activities related to this one",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "required":
["UserDomainID", "EndpointID", "ShareID", "ResourceID", "ActivityID"]
  },
  "required": ["Associations"]
}
```

}

13.3 AdminDomain.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/AdminDomain.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Domain.json"}],
  "properties": {
    "Distributed": {
      "type": ["boolean", "null"],
      "description": "true if the services managed by the AdminDomain are
geographically distributed"
    },
    "Owner": {
      "type": "array",
      "description": "Identification of persons or legal entities that own
the resources in this AdminDomain",
      "items": {
        "type": "string"
      }
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ServiceID": {
          "type": "array",
          "description": "IDs of Services in this AdminDomain",
          "items": {
            "type": "string"
          }
        },
        "ChildDomainID": {
          "type": "array",
          "description": "IDs of AdminDomains aggregated by this
AdminDomain",
          "items": {
            "type": "string"
          }
        },
        "ParentDomainID": {
          "type": ["string", "null"],
          "description": "The ID of the AdminDomain that this AdminDomain
participates in"
        },
        "ComputingServiceID": {
          "type": "array",
          "description": "IDs of ComputingServices in this AdminDomain",
          "items": {
            "type": "string"
          }
        },
        "StorageServiceID": {
          "type": "array",
          "description": "IDs of StorageServices in this AdminDomain",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "required": ["ParentDomainID"]
  },
  "required": ["Associations"]
}

```


13.4 ApplicationEnvironment.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ApplicationEnvironment.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "AppName": {
      "type": "string",
      "description": "The name of the application"
    },
    "AppVersion": {
      "type": "string",
      "description": "The version of the application"
    },
    "State": {
      "type": "string",
      "description": "The current installation state of the application -
AppEnvState_t"
    },
    "RemovalDate": {
      "type": "string",
      "description": "The date/time after which the application may be
removed - DateTime_t"
    },
    "License": {
      "type": "string",
      "description": "The license under which the application is usable -
License_t"
    },
    "Description": {
      "type": "string",
      "description": "A human-readable description of the application"
    },
    "BestBenchmark": {
      "type": "array",
      "description": "The type(s) of the benchmarks which best describe the
sensitivity of this application to the performance of the
ExecutionEnvironment - Benchmark_t",
      "items": {
        "type": "string"
      }
    },
    "ParallelSupport": {
      "type": "string",
      "description": "The type of supported parallel execution -
ParallelSupport_t"
    },
    "MaxSlots": {
      "type": "integer",
      "description": "The maximum number of concurrent slots that may be
used to run the application"
    },
    "MaxJobs": {
      "type": "integer",
      "description": "The maximum number of concurrent jobs that can run the
application"
    },
    "MaxUserSeats": {
      "type": "integer",
      "description": "The maximum number of concurrent users that can run
the application"
    },
    "FreeSlots": {

```

```

    "type": "integer",
    "description": "The maximum number slots currently available to run
the application"
  },
  "FreeJobs": {
    "type": "integer",
    "description": "The maximum number of additional jobs that can run the
application"
  },
  "FreeUserSeats": {
    "type": "integer",
    "description": "The maximum number of additional users that can run
the application"
  },
  "Associations": {
    "type": "object",
    "properties": {
      "ExecutionEnvironmentID": {
        "type": "array",
        "description": "ID(s) of ExecutionEnvironments where this
ApplicationEnvironment can be used",
        "items": {
          "type": "string"
        }
      },
      "ComputingManagerID": {
        "type": "string",
        "description": "ID of the ComputingManager this
ApplicationEnvironment is associated with"
      },
      "ApplicationHandleID": {
        "type": "array",
        "description": "ID(s) of the ApplicationHandles that can be used
to refer to this environment",
        "items": {
          "type": "string"
        }
      }
    },
    "required": ["ExecutionEnvironmentID", "ComputingManagerID"]
  }
},
"required": ["AppName", "Associations"]
}

```

13.5 ApplicationHandle.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ApplicationHandle.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of method used to set up an
ApplicationEnvironment - ApplicationHandle_t (open enumeration)"
    },
    "Value": {
      "type": "string",
      "description": "How to set up the ApplicationEnvironment in the
context of the Type"
    },
    "Associations": {
      "type": "object",

```

```

    "properties": {
      "ApplicationEnvironmentID": {
        "type": "string",
        "description": "The ID of the ApplicationEnvironment this
ApplicationHandle refers to"
      }
    },
    "required": ["ApplicationEnvironmentID"]
  }
},
"required": ["Type", "Value", "Associations"]
}

```

13.6 Benchmark.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Benchmark.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of the benchmark - Benchmark_t (open
enumeration)"
    },
    "Value": {
      "type": "number",
      "description": "The value of the benchmark"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ExecutionEnvironmentID": {
          "type": ["string", "null"],
          "description": "ID of the ExecutionEnvironment this benchmark is
for",
          "items": {
            "type": "string"
          }
        },
        "ComputingManagerID": {
          "type": ["string", "null"],
          "description": "The ID of the ComputingManager this benchmark is
for"
        }
      },
      "required": ["ExecutionEnvironmentID", "ComputingManagerID"]
    }
  },
  "required": ["Type", "Value", "Associations"]
}

```

13.7 ComputingActivity.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingActivity.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Activity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "closed enumeration ComputingActivityType_t",
      "enum":
["collectionelement", "parallelelement", "single", "workflownode"]
    }
  }
}

```

```

},
"IDFromEndpoint": {
  "type": "string",
  "description": "The ID assigned by the ComputingEndpoint"
},
"LocalIDFromManager": {
  "type": "string",
  "description": "The local ID assigned by the ComputingManager"
},
"JobDescription": {
  "type": "string",
  "description": "open enumeration JobDescription_t"
},
"State": {
  "type": "array",
  "description": "open enumeration ComputingActivityState_t",
  "items": {
    "type": "string"
  },
  "minItems": 1
},
"RestartState": {
  "type": "array",
  "description": "open enumeration ComputingActivityState_t",
  "items": {
    "type": "string"
  }
},
"ExitCode": {
  "type": "integer",
  "description": "The exit code as returned by the main executable code
or script of the job"
},
"ComputingManagerExitCode": {
  "type": "string",
  "description": "The exit code provided by the ComputingManager"
},
"Error": {
  "type": "array",
  "description": "The error messages as provided by the software
components involved in the management of the job",
  "items": {
    "type": "string"
  }
},
"WaitingPosition": {
  "type": "integer",
  "description": "The position of the job in the queue, if the job is
waiting"
},
"Owner": {
  "type": "string",
  "description": "The Grid identity of the job's owner"
},
"LocalOwner": {
  "type": "string",
  "description": "The local user name of the job's owner"
},
"RequestedTotalWallTime": {
  "type": "integer",
  "description": "The total wall clock time requested by the job"
},
"RequestedTotalCPUTime": {
  "type": "integer",
  "description": "The total CPU time requested by the job"
},
"RequestedSlots": {
  "type": "integer",

```

```

    "description": "The number of slots requested for the job"
  },
  "RequestedApplicationEnvironment": {
    "type": "array",
    "description": "The AppName and Version of the requested
ApplicationEnvironments",
    "items": {
      "type": "string"
    }
  },
  "StdIn": {
    "type": "string",
    "description": "The name of the file used for standard input"
  },
  "StdOut": {
    "type": "string",
    "description": "The name of the file used for standard output"
  },
  "StdErr": {
    "type": "string",
    "description": "The name of the file used for standard error"
  },
  "LogDir": {
    "type": "string",
    "description": "The name of the directory which contains job logs"
  },
  "ExecutionNode": {
    "type": "array",
    "description": "Hostnames associated with the ExecutionEnvironments
running the job",
    "items": {
      "type": "string"
    }
  },
  "Queue": {
    "type": "string",
    "description": "The name of the ComputingManager queue that held the
job before execution"
  },
  "UsedTotalWallTime": {
    "type": "integer",
    "description": "The total wall clock time consumed by the job so far
(slots*seconds)"
  },
  "UsedTotalCpuTime": {
    "type": "integer",
    "description": "The total CPU time consumed by the job so far
(seconds)"
  },
  "UsedMainMemory": {
    "type": "integer",
    "description": "The physical RAM currently used by the job (MB)"
  },
  "SubmissionTime": {
    "type": "string",
    "description": "The time when the job was submitted to the
ComputingEndpoint (DateTime_t)"
  },
  "ComputingManagerSubmissionTime": {
    "type": "string",
    "description": "The time when the job was submitted to the
ComputingManager (DateTime_t)"
  },
  "StartTime": {
    "type": "string",
    "description": "The time when the ComputingManager started the job
(DateTime_t)"
  },
  },

```

```

    "ComputingManagerEndTime": {
      "type": "string",
      "description": "The time when the job ended according to the
ComputingManager (DateTime_t)"
    },
    "EndTime": {
      "type": "string",
      "description": "The time when the job ended in the Grid layer
(DateTime_t)"
    },
    "WorkingAreaEraseTime": {
      "type": "string",
      "description": "The time when working area will be removed from
storage (DateTime_t)"
    },
    "ProxyExpirationTime": {
      "type": "string",
      "description": "The expiration time of the Grid proxy associated with
the job (DateTime_t)"
    },
    "SubmissionHost": {
      "type": "string",
      "description": "The name of the host from which the job was submitted"
    },
    "SubmissionClientName": {
      "type": "string",
      "description": "The name of the software client used to submit the
job"
    },
    "OtherMessages": {
      "type": "array",
      "description": "Optional messages provided by either the Grid layer or
the ComputingManager",
      "items": {
        "type": "string"
      }
    }
  },
  "required": ["State","Owner"]
}

```

13.8 ComputingEndpoint.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingEndpoint.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Endpoint.json"}],
  "properties": {
    "Staging": {
      "type": "string",
      "description": "Supported file staging functionality - Staging_t",
      "enum": ["none","stagingin","staginginout","stagingout"]
    },
    "JobDescription": {
      "type": "array",
      "description": "Supported job description languages - JobDescription_t
(open Enumeration)",
      "items": {
        "type": "string"
      }
    },
    "TotalJobs": {
      "type": "integer",
      "description": "The total number of Grid jobs known to the system"
    },
    "RunningJobs": {

```

```

    "type": "integer",
    "description": "The number of Grid jobs which are running in an
ExecutionEnvironment"
  },
  "WaitingJobs": {
    "type": "integer",
    "description": "The number of Grid jobs which are waiting to start
executing"
  },
  "StagingJobs": {
    "type": "integer",
    "description": "The number of Grid jobs staging files before or after
execution"
  },
  "SuspendedJobs": {
    "type": "integer",
    "description": "The number of Grid jobs that started to execute, but
are now suspended"
  },
  "PreLRMSWaitingJobs": {
    "type": "integer",
    "description": "The number of Grid jobs managed by the Grid software,
but not yet passed to the LRMS"
  }
}
}

```

13.9 ComputingManager.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingManager.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Manager.json"}],
  "properties": {
    "Reservation": {
      "type": ["boolean", "null"],
      "description": "Whether advance reservation is supported (no value
implies undefined in ExtendedBoolean_t)"
    },
    "BulkSubmission": {
      "type": ["boolean", "null"],
      "description": "Whether multiple jobs can be submitted at once (no
value implies undefined in ExtendedBoolean_t)"
    },
    "TotalPhysicalCPUs": {
      "type": "integer",
      "description": "The total number of physical CPUs managed by this
ComputingManager"
    },
    "TotalLogicalCPUs": {
      "type": "integer",
      "description": "The total number of logical CPUs managed by this
ComputingManager"
    },
    "TotalSlots": {
      "type": "integer",
      "description": "The total number of slots managed by this
ComputingManager"
    },
    "SlotsUsedByLocalJobs": {
      "type": "integer",
      "description": "The number of slots currently used by jobs submitted
via a non-Grid interface"
    },
    "SlotsUsedByGridJobs": {
      "type": "integer",

```

```

    "description": "The number of slots currently used by jobs submitted
via a non-Grid interface"
  },
  "Homogeneous": {
    "type": ["boolean","null"],
    "description": "Whether this ComputingManager manages only one type of
ExecutionEnvironment"
  },
  "NetworkInfo": {
    "type": "array",
    "description": "The types of internal network connections between
ExecutionEnvironments (NetworkInfo_t)",
    "items": {
      "type": "string"
    }
  },
  "LocalCPUDistribution": {
    "type": ["boolean","null"],
    "description": "Classification of the managed ExecutionEnvironments
aggregated by the number of logical CPUs"
  },
  "WorkingAreaShared": {
    "type": ["boolean","null"],
    "description": "True if the working area is shared across different
ExecutionEnvironments"
  },
  "WorkingAreaGuaranteed": {
    "type": ["boolean","null"],
    "description": "True if the job is guaranteed all of WorkingAreaTotal"
  },
  "WorkingAreaTotal": {
    "type": "integer",
    "description": "Total size of the working area available to single
slot jobs (GB)"
  },
  "WorkingAreaFree": {
    "type": "integer",
    "description": "The amount of free space in the working area (GB)"
  },
  "WorkingAreaLifeTime": {
    "type": "integer",
    "description": "The minimum guaranteed lifetime of files created in
the working area (seconds)"
  },
  "WorkingAreaMultiSlotTotal": {
    "type": "integer",
    "description": "The total size of the working area across all
ExecutionEnvironments (GB)"
  },
  "WorkingAreaMultiSlotFree": {
    "type": "integer",
    "description": "The available space in the working area across all
ExecutionEnvironments (GB)"
  },
  "WorkingAreaMultiSlotLifeTime": {
    "type": "integer",
    "description": "The minimum guaranteed lifetime of files created in
the working area (seconds)"
  },
  "CacheTotal": {
    "type": "integer",
    "description": "If local caching of input files is supported, the
total size of the area they may be stored in"
  },
  "CacheFree": {
    "type": "integer",
    "description": "If local caching of input files is supported, the
available size of the area they may be stored in"
  }
}

```



```

    },
    "TmpDir": {
      "type": "string",
      "description": "The absolute path of a temporary directory local to an
ExecutionEnvironment"
    },
    "ScratchDir": {
      "type": "string",
      "description": "The absolute path of a shared directory available for
application data"
    },
    "ApplicationDir": {
      "type": "string",
      "description": "The absolute path of a directory available for
installation of persistent application software"
    },
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ApplicationEnvironmentID": {
          "type": "array",
          "description": "ID(s) of ApplicationEnvironments provided by this
ComputingManager",
          "items": {
            "type": "string"
          }
        },
        "BenchmarkID": {
          "type": "array",
          "description": "ID(s) of Benchmarks associated with this
ComputingManager",
          "items": {
            "type": "string"
          }
        }
      }
    }
  }
}

```

13.10 ComputingService.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingService.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Service.json"}],
  "properties": {
    "TotalJobs": {
      "type": "integer",
      "description": "The total number of Grid jobs known to the system"
    },
    "RunningJobs": {
      "type": "integer",
      "description": "The number of Grid jobs which are running in an
ExecutionEnvironment"
    },
    "WaitingJobs": {
      "type": "integer",
      "description": "The number of Grid jobs which are waiting to start
executing"
    },
    "StagingJobs": {
      "type": "integer",
      "description": "The number of Grid jobs staging files before or after
execution"
    }
  },
}

```

```

    "SuspendedJobs": {
      "type": "integer",
      "description": "The number of Grid jobs that started to execute, but
are now suspended"
    },
    "PreLRMSWaitingJobs": {
      "type": "integer",
      "description": "The number of Grid jobs managed by the Grid software,
but not yet passed to the LRMS"
    }
  }
}

```

13.11 ComputingShare.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingShare.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Share.json"}],
  "properties": {
    "MappingQueue": {
      "type": "string",
      "description": "The name of the queue in the LRMS where jobs in this
share are submitted"
    },
    "MaxWallTime": {
      "type": "integer",
      "description": "The maximum wall clock time that a single-slot job can
run (seconds)"
    },
    "MaxMultiSlotWallTime": {
      "type": "integer",
      "description": "The maximum wall clock time that a multi-slot job can
run (seconds)"
    },
    "DefaultWallTime": {
      "type": "integer",
      "description": "The default wall clock per slot assumed by the LRMS if
a maximum time is not specified (seconds)"
    },
    "MaxCPUTime": {
      "type": "integer",
      "description": "The maximum pre-slot CPU time that a job can request
(seconds)"
    },
    "MaxTotalCPUTime": {
      "type": "integer",
      "description": "The maximum amount of CPU time that a job can request
across all slots assigned to it (seconds)"
    },
    "MinCPUTime": {
      "type": "integer",
      "description": "The minimum pre-slot CPU time that a job can request
(seconds)"
    },
    "DefaultCPUTime": {
      "type": "integer",
      "description": "The default CPU time limit assumed by the LRMS if a
maximum time is not specified (seconds)"
    },
    "MaxTotalJobs": {
      "type": "integer",
      "description": "The maximum number of jobs that can be in this Share"
    },
    "MaxRunningJobs": {
      "type": "integer",

```

```

    "description": "The maximum number of jobs that can be running in this
Share"
  },
  "MaxWaitingJobs": {
    "type": "integer",
    "description": "The maximum number of jobs that can be waiting in this
Share"
  },
  "MaxPreLRMSWaitingJobs": {
    "type": "integer",
    "description": "The maximum number of jobs that can be waiting in the
Grid layer for this Share"
  },
  "MaxUserRunningJobs": {
    "type": "integer",
    "description": "The maximum number of jobs that can be running in this
Share per user"
  },
  "MaxSlotsPerJob": {
    "type": "integer",
    "description": "The maximum number of slots that can be allocated to a
single job in this Share"
  },
  "MaxStageInStreams": {
    "type": "integer",
    "description": "The maximum number of streams available to stage files
in"
  },
  "MaxStageOutStreams": {
    "type": "integer",
    "description": "The maximum number of streams available to stage files
out"
  },
  "SchedulingPolicy": {
    "type": "string",
    "description": "The scheduling policy used by the share -
SchedulingPolicy_t (open enumeration)"
  },
  "MaxMainMemory": {
    "type": "integer",
    "description": "The maximum amount of physical RAM that a job can use
(MB)"
  },
  "GuaranteedMainMemory": {
    "type": "integer",
    "description": "The amount of physical RAM that a job will have
available (MB)"
  },
  "MaxVirtualMemory": {
    "type": "integer",
    "description": "The maximum amount memory (RAM+swap) that a job can
use (MB)"
  },
  "GuaranteedVirtualMemory": {
    "type": "integer",
    "description": "The amount of memory (RAM+swap) that a job will have
available (MB)"
  },
  "MaxDiskSpace": {
    "type": "integer",
    "description": "The maximum disk space that a job can use in the
working area (GB)"
  },
  "DefaultStorageService": {
    "type": "string",
    "description": "The ID of the default StorageService used to store
files"
  },
  },

```

```

"Preemption": {
  "type": ["boolean","null"],
  "description": "Whether jobs can be preempted and resumed (no value
implies undefined in ExtendedBoolean_t)"
},
"ServingState": {
  "type": "string",
  "description": "How the Share is currently serving jobs",
  "enum": ["closed","draining","production","queueing"]
},
},
"TotalJobs": {
  "type": "integer",
  "description": "The total number of jobs in any state"
},
},
"RunningJobs": {
  "type": "integer",
  "description": "The number of running jobs submitted through Grid or
non-Grid interfaces"
},
},
"LocalRunningJobs": {
  "type": "integer",
  "description": "The number of running jobs submitted using non-Grid
interfaces"
},
},
"WaitingJobs": {
  "type": "integer",
  "description": "The number of waiting jobs submitted through Grid or
non-Grid interfaces"
},
},
"LocalWaitingJobs": {
  "type": "integer",
  "description": "The number of waiting jobs submitted using non-Grid
interfaces"
},
},
"SuspendedJobs": {
  "type": "integer",
  "description": "The number of suspended jobs submitted through Grid or
non-Grid interfaces"
},
},
"LocalSuspendedJobs": {
  "type": "integer",
  "description": "The number of suspended jobs submitted using non-Grid
interfaces"
},
},
"StagingJobs": {
  "type": "integer",
  "description": "The number of jobs staging files before or after
execution"
},
},
"PreLRMSWaitingJobs": {
  "type": "integer",
  "description": "The number of Grid jobs which have not yet been passed
to the LRMS"
},
},
"EstimatedAverageWaitingTime": {
  "type": "integer",
  "description": "An estimate of the average time a job will wait before
it starts to execute (seconds)"
},
},
"EstimatedWorstWaitingTime": {
  "type": "integer",
  "description": "An estimate of the worst-case time a job will wait
before it starts to execute (seconds)"
},
},
"FreeSlots": {
  "type": "integer",
  "description": "The number of slots which are currently available for
use"
}

```

```

    },
    "FreeSlotsWithDuration": {
      "type": "string",
      "description": "The number of slots which are currently available for
use and how long they are available"
    },
    "UsedSlots": {
      "type": "integer",
      "description": "The number of slots currently in use"
    },
    "RequestedSlots": {
      "type": "integer",
      "description": "The number of slots needd to execute all waiting and
staging jobs"
    },
    "ReservationPolicy": {
      "type": "string",
      "description": "The policy used for advance reservation -
ReservationPolicy_t",
      "enum": ["mandatory","none","optional"]
    },
    "Tag": {
      "type": "array",
      "description": "UserDomain-defined tags for this Share",
      "items": {
        "type": "string"
      }
    }
  },
  "required": ["ServingState"]
}

```

13.12 Contact.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Contact.json",
  "description": "A GLUE 2 Contact",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Detail": {
      "type": "string",
      "description": "A URI embedding the contact information"
    },
    "Type": {
      "type": "string",
      "description": "closed enumeration ContactType_t",
      "enum": ["general","security","sysadmin","usersupport"]
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ServiceID": {
          "type": "array",
          "description": "The IDs of Services associated with this Contact",
          "items": {
            "type": "string"
          }
        },
        "DomainID": {
          "type": "array",
          "description": "The IDs of Domains associated with this Contact",
          "items": {
            "type": "string"
          }
        }
      }
    }
  }
},

```

```

    "required": ["DomainID"]
  }
},
"required": ["Associations"]
}

```

13.13 DataStore.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/DataStore.json",
  "type": "object",
  "allOf": [{"$ref": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Resource.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of storage medium - DataStoreType_t (disk,optical,tape,...)"
    },
    "Latency": {
      "type": "string",
      "description": "The latency category under normal operating conditions - AccessLatency_t",
      "enum": ["nearline","offline","online"]
    },
    "TotalSize": {
      "type": "integer",
      "description": "The total amount of storage (GB)"
    },
    "FreeSize": {
      "type": "integer",
      "description": "The amount of available storage (GB)"
    },
    "UsedSize": {
      "type": "integer",
      "description": "The amount of used storage (GB)"
    }
  },
  "required": ["Type","Latency"]
}

```

13.14 Domain.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Domain.json",
  "type": "object",
  "allOf": [{"$ref": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Description": {
      "type": "string",
      "description": "A description of the Domain"
    },
    "WWW": {
      "type": "array",
      "description": "URLs of web pages with more information about the Domain",
      "items": {
        "type": "string"
      }
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ContactID": {
          "type": "array",
          "description": "IDs of Contacts for this Domain",

```

```

        "items": {
            "type": "string"
        }
    },
    "LocationID": {
        "type": ["string", "null"],
        "description": "The ID of the primary Location for this Domain"
    }
},
"required": ["LocationID"]
}
},
"required": ["Associations"]
}

```

13.15 Endpoint.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Endpoint.json",
  "type": "object",
  "allOf": [{"$ref": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "URL": {
      "type": "string",
      "description": "Network location of the endpoint"
    },
    "Capability": {
      "type": "array",
      "description": "Capability_t (open enumeration)",
      "items": {
        "type": "string"
      }
    },
    "Technology": {
      "type": "string",
      "description": "EndpointTechnology_t"
    },
    "InterfaceName": {
      "type": "string",
      "description": "InterfaceName_t"
    },
    "InterfaceVersion": {
      "type": "string",
      "description": "The version of the primary interface protocol (free format)"
    },
    "InterfaceExtension": {
      "type": "array",
      "description": "URIs identifying supported extensions to the interface protocol",
      "items": {
        "type": "string"
      }
    },
    "WSDL": {
      "type": "array",
      "description": "URLs of WSDL document(s) describing the interface",
      "items": {
        "type": "string"
      }
    },
    "SupportedProfile": {
      "type": "array",
      "description": "URI(s) identifying supported profiles for the Endpoint",
      "items": {
        "type": "string"
      }
    }
  }
}

```

```

    }
  },
  "Semantics": {
    "type": "array",
    "description": "URL(s) of documents providing human-readable
descriptions of the semantics of the Endpoint",
    "items": {
      "type": "string"
    }
  },
  "Implementor": {
    "type": "string",
    "description": "The name of the main organization implementing the
Endpoint"
  },
  "ImplementationName": {
    "type": "string",
    "description": "The name of the implementation of the Endpoint"
  },
  "ImplementationVersion": {
    "type": "string",
    "description": "The version of the implementation of the Endpoint"
  },
  "QualityLevel": {
    "type": "string",
    "description": "QualityLevel_t",
    "enum": ["development", "pre-production", "production", "testing"]
  },
  "HealthState": {
    "type": "string",
    "description": "The operational status of the Endpoint",
    "enum": ["critical", "ok", "other", "unknown", "warning"]
  },
  "HealthStateInfo": {
    "type": "string",
    "description": "A human-readable explanation of the HealthState of
this Endpoint"
  },
  "ServingState": {
    "type": "string",
    "description": "If the endpoint is accepting and serving requests",
    "enum": ["closed", "draining", "production", "queueing"]
  },
  "StartTime": {
    "type": "string",
    "description": "The start time of the Service associated with this
Endpoint (DateTime_t)"
  },
  "IssuerCA": {
    "type": "string",
    "description": "The DN of the CA issuing the certificate presented by
this Endpoint"
  },
  "TrustedCA": {
    "type": "array",
    "description": "DN(s) of CAs trusted by this Endpoint",
    "items": {
      "type": "string"
    }
  },
  "DowntimeAnnounce": {
    "type": "string",
    "description": "When the next scheduled downtime was announced
(DateTime_t)"
  },
  "DowntimeStart": {
    "type": "string",
    "description": "When the next scheduled downtime will start

```



```

(DateTime_t)"
  },
  "DowntimeEnd": {
    "type": "string",
    "description": "When the next scheduled downtime will end
(DateTime_t)"
  },
  "DowntimeInfo": {
    "type": "string",
    "description": "Human-readable of the next scheduled downtime"
  },
  "Associations": {
    "type": "object",
    "properties": {
      "ServiceID": {
        "type": "string",
        "description": "The ID of the Service associated with this
Endpoint"
      },
      "ShareID": {
        "type": "array",
        "description": "The IDs of the Shares accessible from this
Endpoint",
        "items": {
          "type": "string"
        }
      },
      "AccessPolicyID": {
        "type": "array",
        "description": "IDs of AccessPolicies associated with this
Endpoint",
        "items": {
          "type": "string"
        }
      },
      "ActivityID": {
        "type": "array",
        "description": "IDs of Activities being managed through this
Endpoint",
        "items": {
          "type": "string"
        }
      }
    },
    "required": ["ServiceID","ShareID"]
  }
},
"required":
["InterfaceName","QualityLevel","HealthState","ServingState","Associations"]
}

```

13.16 Entity.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json",
  "type": "object",
  "properties": {
    "CreationTime": {
      "type": "string",
      "description": "The creation time of this entity in the format: CCYY-
MM-DDThh:mm:ss[Z|(+|-)hh:mm]"
    },
    "Validity": {
      "type": "integer",
      "description": "The number of seconds after CreationTime that this
entity should be considered relevant"
    },
    "ID": {

```

```

    "type": "string",
    "description": "A globally unique identifier for this entity"
  },
  "Name": {
    "type": "string",
    "description": "A human-readable name"
  },
  "OtherInfo": {
    "type": "array",
    "description": "Placeholder for information that does not fit in any
other attribute",
    "items": {
      "type": "string"
    }
  },
  "Extension": {
    "type": "object",
    "description": "Key/value pairs enabling the association of extra
information not captured by the model"
  }
},
"required": ["ID"]
}

```

13.17 ExecutionEnvironment.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ExecutionEnvironment.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Resource.json"}],
  "properties": {
    "Platform": {
      "type": "string",
      "description": "The platform architecture - Platform_t (open
enumeration)"
    },
    "VirtualMachine": {
      "type": ["boolean", "null"],
      "description": "True if the ExecutionEnvironment is a virtual machine"
    },
    "TotalInstances": {
      "type": "integer",
      "description": "The total number of ExecutionEnvironment instances"
    },
    "UsedInstances": {
      "type": "integer",
      "description": "The number of ExecutionEnvironment instances in use"
    },
    "UnavailableInstances": {
      "type": "integer",
      "description": "The number of ExecutionEnvironment instances that are
unavailable"
    },
    "PhysicalCPUs": {
      "type": "integer",
      "description": "The number of physical CPUs in one
ExecutionEnvironment instance"
    },
    "LogicalCPUs": {
      "type": "integer",
      "description": "The number of logical CPUs in one ExecutionEnvironment
instance"
    },
    "CPUMultiplicity": {
      "type": "string",
      "description": "Information about the CPUs and cores in an execution

```

```

environment",
  "enum": ["multicpu-multicore","multicpu-singlecore","singlecpu-
multicore","singlecpu-singlecore"]
},
"CPUVendor": {
  "type": "string",
  "description": "The name of the manufacturer of the CPU"
},
"CPUModel": {
  "type": "string",
  "description": "The model of the CPU, as defined by the vendor"
},
"CPUVersion": {
  "type": "string",
  "description": "The specific version name of the CPU, as defined by
the vendor"
},
"CPUClockSpeed": {
  "type": "integer",
  "description": "The clock speed of the CPU (MHz)"
},
"CPUTimeScalingFactor": {
  "type": "float",
  "description": "The factor used by the ComputingManager to scale the
CPU time limit"
},
"WallTimeScalingFactor": {
  "type": "float",
  "description": "The factor used by the ComputingManager to scale the
wallclock time limit"
},
"MainMemorySize": {
  "type": "integer",
  "description": "The total amount of physical RAM in one
ExecutionEnvironment instance (MB)"
},
"VirtualMemorySize": {
  "type": "integer",
  "description": "The total amount of virtual memory (RAM+swap) in one
ExecutionEnvironment instance (MB)"
},
"OSFamily": {
  "type": "string",
  "description": "The general family of the operating system -
OSFamily_t (open enumeration)"
},
"OSName": {
  "type": "string",
  "description": "The specific name of the operating system - OSName_t
(open enumeration)"
},
"OSVersion": {
  "type": "string",
  "description": "The version of the operating system, as defined by the
vendor"
},
"ConnectivityIn": {
  "type": ["boolean","null"],
  "description": "True if direct inbound network connectiity is
available to a running job"
},
"ConnectivityOut": {
  "type": ["boolean","null"],
  "description": "True if direct outbound network connectiity is
available to a running job"
},
"NetworkInfo": {
  "type": "array",

```

```

    "description": "The types of internal network connections between
ExecutionEnvironments - NetworkInfo_t (open enumeration)",
    "items": {
        "type": "string"
    }
},
"Associations": {
    "type": "object",
    "properties": {
        "ApplicationEnvironmentID": {
            "type": "array",
            "description": "ID(s) of ApplicationEnvironments available in this
ExecutionEnvironment",
            "items": {
                "type": "string"
            }
        },
        "BenchmarkID": {
            "type": "array",
            "description": "ID(s) of Benchmarks associated with this
ExecutionEnvironment",
            "items": {
                "type": "string"
            }
        }
    }
}
},
"required":
["Platform", "MainMemorySize", "OSFamily", "ConnectivityIn", "ConnectivityOut"]
}

```

13.18 Glue2.json

```

{
    "$schema": "http://json-schema.org/draft-04/schema#",
    "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Glue2.json",
    "description": "A GLUE 2 document",
    "type": "object",
    "properties": {
        "Entity": {
            "type": "array",
            "items": {
                "type": "object",
                "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}]
            }
        },
        "Location": {
            "type": "array",
            "items": {
                "type": "object",
                "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Location.json"}]
            }
        },
        "Contact": {
            "type": "array",
            "items": {
                "type": "object",
                "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Contact.json"}]
            }
        },
        "Domain": {
            "type": "array",
            "items": {
                "type": "object",
                "allOf": [{"$ref":

```

```

"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Domain.json"]
  }
},
"AdminDomain": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/AdminDomain.json"}]
  }
},
"UserDomain": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/UserDomain.json"}]
  }
},
"Service": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Service.json"}]
  }
},
"Endpoint": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Endpoint.json"}]
  }
},
"Share": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Share.json"}]
  }
},
"Manager": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Manager.json"}]
  }
},
"Resource": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Resource.json"}]
  }
},
"Activity": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Activity.json"}]
  }
},
"Policy": {

```

```

    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Policy.json"}]
    }
  },
  "AccessPolicy": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/AccessPolicy.json"}]
    }
  },
  "MappingPolicy": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/MappingPolicy.json"}]
    }
  },
  "ComputingService": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingService.json"}]
    }
  },
  "ComputingEndpoint": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingEndpoint.json"}]
    }
  },
  "ComputingShare": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingShare.json"}]
    }
  },
  "ComputingManager": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingManager.json"}]
    }
  },
  "Benchmark": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Benchmark.json"}]
    }
  },
  "ExecutionEnvironment": {
    "type": "array",
    "items": {
      "type": "object",
      "allOf": [{"$ref":

```

```

"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ExecutionEnvironment.json"}
]
}
},
"ApplicationEnvironment": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ApplicationEnvironment.json"}]}
}
},
"ApplicationHandle": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ApplicationHandle.json"}]}
}
},
"ComputingActivity": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ComputingActivity.json"}]}
}
},
"ToStorageService": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ToStorageService.json"}]}
}
},
"StorageService": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageService.json"}]}
}
},
"StorageServiceCapacity": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageServiceCapacity.json"}]}
}
},
"StorageAccessProtocol": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageAccessProtocol.json"}]}
}
},
"StorageEndpoint": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":

```

```

"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageEndpoint.json"]
  }
},
"StorageShare": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageShare.json"}]
  }
},
"StorageShareCapacity": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageShareCapacity.json"}]
}
},
"StorageManager": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageManager.json"}]
  }
},
"DataStore": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/DataStore.json"}]
  }
},
"ToComputingService": {
  "type": "array",
  "items": {
    "type": "object",
    "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ToComputingService.json"}]
  }
}
}
}

```

13.19 Location.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Location.json",
  "description": "A GLUE 2 Location",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Address": {
      "type": "string",
      "description": "A free format street address"
    },
    "Place": {
      "type": "string",
      "description": "Name of town/city"
    },
    "Country": {
      "type": "string",
      "description": "Name of country"
    }
  }
}

```



```

    "PostalCode": {
      "type": "string",
      "description": "Postal code"
    },
    "Latitude": {
      "type": "number",
      "description": "Position north (positive) or south (negative) of the
equator in degrees"
    },
    "Longitude": {
      "type": "number",
      "description": "Position east (positive) or west (negative) of the
primary meridian in degrees"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ServiceID": {
          "type": "array",
          "description": "The IDs of Services at this location",
          "items": {
            "type": "string"
          }
        },
        "DomainID": {
          "type": "array",
          "description": "The IDs of Domains at this location",
          "items": {
            "type": "string"
          }
        }
      }
    }
  }
}

```

13.20 Manager.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Manager.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "ProductName": {
      "type": "string",
      "description": "The name of the software product which implements the
Manager"
    },
    "ProductVersion": {
      "type": "string",
      "description": "The version of the software product which implements
the Manager"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ServiceID": {
          "type": "string",
          "description": "The ID of the Service this Manager participates in"
        },
        "ResourceID": {
          "type": "array",
          "description": "ID(s) of Resources associated with this Manager",
          "items": {
            "type": "string"
          }
        }
      },
      "minItems": 1
    }
  }
}

```

```

    }
  },
  "required": ["ServiceID","ResourceID"]
}
},
"required": ["ProductName","Associations"]
}

```

13.21 MappingPolicy.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/MappingPolicy.json",
  "type": "object",
  "allOf": [{"$ref": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Policy.json"}],
  "properties": {
    "Associations": {
      "type": "object",
      "properties": {
        "ShareID": {
          "type": "string",
          "description": "The ID of the Share this MappingPolicy is for"
        }
      },
      "required": ["ShareID"]
    }
  }
}

```

13.22 Policy.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Policy.json",
  "type": "object",
  "allOf": [{"$ref": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Scheme": {
      "type": "string",
      "description": "PolicyScheme_t (open enumeration)"
    },
    "Rule": {
      "type": "array",
      "description": "Policy rules",
      "items": {
        "type": "string"
      },
      "minItems": 1
    },
    "Associations": {
      "type": "object",
      "properties": {
        "UserDomainID": {
          "type": "array",
          "description": "The ID(s) of the UserDomains this Policy applies to",
          "items": {
            "type": "string"
          },
          "minItems": 1
        }
      },
      "required": ["UserDomainID"]
    }
  },
  "required": ["Scheme","Rule","Associations"]
}

```

}

13.23 Resource.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Resource.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Associations": {
      "type": "object",
      "properties": {
        "ManagerID": {
          "type": "string",
          "description": "The ID of the Manager for this Resource"
        },
        "ShareID": {
          "type": "array",
          "description": "The ID(s) of the Shares this Resource is part of",
          "items": {
            "type": "string"
          }
        },
        "ActivityID": {
          "type": "array",
          "description": "The ID(s) of Activities consuming from this Share",
          "items": {
            "type": "string"
          }
        }
      },
      "required": ["ManagerID"]
    }
  }
}
```

13.24 Service.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Service.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Capability": {
      "type": "array",
      "description": "Capability_t (open enumeration)",
      "items": {
        "type": "string"
      }
    },
    "Type": {
      "type": "string",
      "description": "ServiceType_t (open enumeration)"
    },
    "QualityLevel": {
      "type": "string",
      "description": "QualityLevel_t",
      "enum": ["development", "pre-production", "production", "testing"]
    },
    "StatusInfo": {
      "type": "array",
      "description": "URLs of web pages providing additional information",
      "items": {
        "type": "string"
      }
    }
  }
}
```

```

    },
    "Complexity": {
      "type": "string",
      "description": "A human-readable description of the number of endpoint
types, shares, and resources"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "EndpointID": {
          "type": "array",
          "description": "The IDs of Endpoints for this Service",
          "items": {
            "type": "string"
          }
        },
        "ShareID": {
          "type": "array",
          "description": "The IDs of the Shares offered by this Service",
          "items": {
            "type": "string"
          }
        },
        "ManagerID": {
          "type": "array",
          "description": "The IDs of the Managers of this Service",
          "items": {
            "type": "string"
          }
        },
        "ContactID": {
          "type": "array",
          "description": "The IDs of Contacts for this Service",
          "items": {
            "type": "string"
          }
        },
        "LocationID": {
          "type": ["string", "null"],
          "description": "The ID of the primary Location of this Service"
        },
        "ServiceID": {
          "type": "array",
          "description": "The IDs of Services related to this Service",
          "items": {
            "type": "string"
          }
        }
      },
      "required": ["ContactID", "LocationID", "ServiceID"]
    },
    "required": ["Type", "QualityLevel", "Associations"]
  }
}

```

13.25 Share.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Share.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Description": {
      "type": "string",
      "description": "A human-readable description of the Share"
    },
    "Associations": {

```

```

    "type": "object",
    "properties": {
      "EndpointID": {
        "type": "array",
        "description": "The ID(s) of the Endpoints that can be used to
access this Share",
        "items": {
          "type": "string"
        }
      },
      "ResourceID": {
        "type": "array",
        "description": "The ID(s) of the Resources associated with this
Share",
        "items": {
          "type": "string"
        }
      },
      "ServiceID": {
        "type": "string",
        "description": "The ID of the Service this Share participates in"
      },
      "ActivityID": {
        "type": "array",
        "description": "The ID(s) of Activities consuming from this Share",
        "items": {
          "type": "string"
        }
      },
      "MappingPolicyID": {
        "type": "array",
        "description": "ID(s) of MappingPolicies associated with this
Share",
        "items": {
          "type": "string"
        }
      }
    },
    "required": ["ResourceID","ServiceID"]
  }
},
"required": ["Associations"]
}

```

13.26 StorageAccessProtocol.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageAccessProtocol.json"
,
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of the protocol - StorageAccessProtocol_t"
    },
    "Version": {
      "type": "string",
      "description": "The version of the protocol supported"
    },
    "MaxStreams": {
      "type": "integer",
      "description": "The maximum number of parallel network streams which
can be usef for a single transfer"
    },
    "Associations": {

```

```

    "type": "object",
    "properties": {
      "StorageServiceID": {
        "type": "string",
        "description": "The ID of the StorageService this protocol is
available for"
      },
      "ToComputingServiceID": {
        "type": "array",
        "description": "The ID(s) ToComputingService objects that describe
connectivity to ComputingServices",
        "items": {
          "type": "string"
        }
      }
    },
    "required": ["StorageServiceID"]
  },
  "required": ["Type", "Version", "Associations"]
}

```

13.27 StorageEndpoint.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageEndpoint.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Endpoint.json"}],
  "properties": {
  }
}

```

13.28 StorageManager.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageManager.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Share.json"}],
  "properties": {
  }
}

```

13.29 StorageService.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageService.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Service.json"}],
  "properties": {
    "Associations": {
      "type": "object",
      "properties": {
        "StorageAccessProtocolID": {
          "type": "array",
          "description": "The ID(s) of the StorageAccessProtocols supported
by this service",
          "items": {
            "type": "string"
          }
        }
      }
    },
    "StorageServiceCapacityID": {

```

```

        "type": "array",
        "description": "The ID(s) of the StorageServiceCapacities for this
Service",
        "items": {
            "type": "string"
        }
    }
}
}

```

13.30 StorageServiceCapacity.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageServiceCapacity.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of storage capacity - StorageCapacity_t"
    },
    "TotalSize": {
      "type": "integer",
      "description": "The total amount of storage of this type (GB)"
    },
    "FreeSize": {
      "type": "integer",
      "description": "The amount of currently available storage of this type
(GB)"
    },
    "UsedSize": {
      "type": "integer",
      "description": "The amount storage of this type in use (GB)"
    },
    "ReservedSize": {
      "type": "integer",
      "description": "The amount storage of this type which is not in use,
but has been reserved for use in use (GB)"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "StorageServiceID": {
          "type": "string",
          "description": "The ID of the StorageService this capacity
describes"
        }
      },
      "required": ["StorageServiceID"]
    }
  },
  "required": ["Type", "Associations"]
}

```

13.31 StorageShare.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageShare.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Share.json"}],
  "properties": {

```

```

"-servingState": {
  "type": "string",
  "description": "How the Share is currently serving jobs",
  "enum": ["closed", "draining", "production", "queueing"]
},
"Path": {
  "type": "string",
  "description": "A default namespace where files are logically placed
when they are stored in this Share"
},
"AccessMode": {
  "type": "array",
  "description": "The type of access allowed for this share -
AccessMode_t (undefined)",
  "items": {
    "type": "string"
  }
},
"SharingID": {
  "type": "string",
  "description": "A local identifier common to the set of StorageShares
which use the same underlying extents"
},
"AccessLatency": {
  "type": "string",
  "description": "The maximum latency category under normal operating
conditions",
  "enum": ["nearline", "offline", "online"]
},
"RetentionPolicy": {
  "type": "string",
  "description": "The quality of data retention - RetentionPolicy_t"
},
"ExpirationMode": {
  "type": "array",
  "description": "Supported file lifetime modes",
  "items": {
    "type": "string",
    "enum": ["neverexpire", "releasewhenexpired", "warnwhenexpired"]
  }
},
"DefaultLifeTime": {
  "type": "integer",
  "description": "The default lifetime assigned to a new file"
},
"MaximumLifeTime": {
  "type": "integer",
  "description": "The maximum lifetime that can be requested for a file"
},
"Tag": {
  "type": "string",
  "description": "An identifier defined by a UserDomain"
},
"Associations": {
  "type": "object",
  "properties": {
    "StorageShareCapacityID": {
      "type": "array",
      "description": "ID of the StorageShareCapacities associated with
this share",
      "items": {
        "type": "string"
      }
    }
  }
}
},
"required": ["ServingState", "SharingID", "AccessLatency"]

```


}

13.32 StorageShareCapacity.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/StorageShareCapacity.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "Type": {
      "type": "string",
      "description": "The type of storage capacity - StorageCapacity_t"
    },
    "TotalSize": {
      "type": "integer",
      "description": "The total amount of storage (GB)"
    },
    "FreeSize": {
      "type": "integer",
      "description": "The amount of available storage (GB)"
    },
    "UsedSize": {
      "type": "integer",
      "description": "The amount of used storage (GB)"
    },
    "ReservedSize": {
      "type": "integer",
      "description": "The amount storage which is not occupied, but has been
reserved for use (GB)"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "StorageShareID": {
          "type": "string",
          "description": "The ID of the StorageShare related to this
capacity"
        }
      },
      "required": ["StorageShareID"]
    }
  },
  "required": ["Type", "Associations"]
}
```

13.33 ToComputingService.json

```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ToComputingService.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "NetworkInfo": {
      "type": "string",
      "description": "Type of network connections between the Storage and
Computing services (NetworkInfo_t)"
    },
    "Bandwidth": {
      "type": "integer",
      "description": "The normal bandwidth available between the Storage and
Computing services (Mb/s)"
    }
  }
}
```

```

    },
    "Associations": {
      "type": "object",
      "properties": {
        "StorageAccessProtocolID": {
          "type": "array",
          "description": "IDs of the protocols that can be used to access the
StorageService",
          "items": {
            "type": "string"
          }
        },
        "ComputingServiceID": {
          "type": "string",
          "description": "The ID of the ComputingService"
        },
        "StorageServiceID": {
          "type": "string",
          "description": "The ID of the StorageService"
        }
      },
      "required": ["ComputingServiceID", "StorageServiceID"]
    },
    "required": ["Associations"]
  }
}

```

13.34 ToStorageService.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/ToStorageService.json",
  "type": "object",
  "allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Entity.json"}],
  "properties": {
    "LocalPath": {
      "type": "string",
      "description": "The path within the ComputingService that is used to
access the StorageService"
    },
    "RemotePath": {
      "type": "string",
      "description": "The path in the StorageService which is associated
with the LocalPath"
    },
    "Associations": {
      "type": "object",
      "properties": {
        "ComputingServiceID": {
          "type": "string",
          "description": "The ID of the ComputingService"
        },
        "StorageServiceID": {
          "type": "string",
          "description": "The ID of the StorageService"
        }
      },
      "required": ["ComputingServiceID", "StorageServiceID"]
    }
  },
  "required": ["LocalPath", "RemotePath", "Associations"]
}

```

13.35 UserDomain.json

```

{
  "$schema": "http://json-schema.org/draft-04/schema#",

```

```

"id": "http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/UserDomain.json",
"type": "object",
"allOf": [{"$ref":
"http://schemas.ogf.org/glue/2014/11/spec_2.0_r1/Domain.json"}],
"properties": {
  "Level": {
    "type": "integer",
    "description": "the number of hops to reach the root of the hierarchy
of UserDomains"
  },
  "userManager": {
    "type": "array",
    "description": "ID for the Endpoint of a Service managing users in
this UserDomain",
    "items": {
      "type": "string"
    }
  },
  "Member": {
    "type": "array",
    "description": "Identifiers for users in this UserDomain",
    "items": {
      "type": "string"
    }
  },
  "Associations": {
    "type": "object",
    "properties": {
      "PolicyID": {
        "type": "array",
        "description": "IDs for Policies associated with this UserDomain",
        "items": {
          "type": "string"
        }
      },
      "ChildDomainID": {
        "type": "array",
        "description": "IDs of UserDomains aggregated by this UserDomain",
        "items": {
          "type": "string"
        }
      },
      "ParentDomainID": {
        "type": "string",
        "description": "The ID of the UserDomain that this UserDomain
participates in"
      },
      "AccessPolicyID": {
        "type": "array",
        "description": "IDs of AccessPolicies associated with this
UserDomain",
        "items": {
          "type": "string"
        }
      },
      "MappingPolicyID": {
        "type": "array",
        "description": "IDs of MappingPolicies associated with this
UserDomain",
        "items": {
          "type": "string"
        }
      }
    }
  }
}

```

